

Ingeniería de software para sistemas embebidos, sistemas críticos e Internet de las Cosas

Irrazábal, Emanuel; Bernal, Rubén; Ríos, José; Mascheroni, Maximiliano Agustín; Sambrana, Iván; Alonso, José Manuel ; Acevedo, Joaquín

Departamento de Informática. Facultad de Ciencias Exactas y Naturales
y Agrimensura. Universidad Nacional del Nordeste
eirrazabal@exa.unne.edu.ar

Resumen

Esta línea de investigación aborda temas de ingeniería del software desde hace tres años, tanto en sistemas tradicionales y su traslado a sistemas embebidos y sistemas críticos. Se espera la aplicación de la tecnología de desarrollo software en sistemas embebidos para el ámbito regional. En este sentido se ha realizado la construcción de un modelo de procedimientos para la gestión de requerimientos en entidades agrícolas con una cultura organizacional jerárquica, específicamente las entidades yerbateras del nordeste argentino. Por otro lado en el estudio de arquitecturas software para el desarrollo de sistemas críticos con aplicación al ámbito ferroviario nacional. Asimismo se está trabajando en el desarrollo de sistemas embebidos con tecnología de comunicación LoRa y la construcción de prototipos para la prevención de incendios forestales.

Palabras clave: Ingeniería de software, sistemas embebidos, sistemas críticos

Contexto

La línea de Investigación y Desarrollo presentada en este trabajo corresponde al proyecto PI-F17-2017 “Análisis e implementación de tecnologías emergentes en sistemas computacionales de aplicación regional.”, acreditado por la Secretaría de Ciencia y Técnica de la Universidad

Nacional del Nordeste (UNNE) para el periodo 2018-2021.

Introducción

Ingeniería de software para sistemas embebidos y sistemas críticos

Un sistema embebido es aquel sistema basado por lo general en un microprocesador, sensores y actuadores diseñado para realizar funciones dedicadas [1]. Y han cobrado gran importancia desde el punto de vista de los sistemas de información con el uso de plataformas tipo Arduino para el desarrollo rápido de prototipos [2].

En este sentido, el grupo de investigación desarrolla de soluciones para entidades regionales, en particular para la agricultura, las entidades públicas provinciales y los grupos de investigación de la universidad.

Uno de los objetivos para este año es la innovación en tecnología de incendios forestales. En este sentido, el nordeste tiene una alta incidencia de incendios; se estima que anualmente se queman aproximadamente 2.000.000 ha. Es común la quema en parches de grandes extensiones próximas a aguadas. Quemadas mal manejadas que frecuentemente alcanzan a los bosques nativos y cultivados [3].

Las labores para el manejo y control del fuego forestal incluyen la detección como un primer paso. [4].

La importancia de lograr la detección temprana de incendios forestales (DTIF) radica en su fácil eliminación a un bajo costo, antes de que se quemen grandes extensiones de bosque. La DTIF está concebida para dar la alarma cuando se ha iniciado el fuego. Por esta razón es necesario utilizar sensores para las principales variables indicadoras de la existencia del fuego: la variable química (emisiones de humo, CO, CO₂, CH₄) y la variable física (radiación infrarroja y el calor) [5].

A su vez, las grandes extensiones de campo hacen imposible el uso de conectividad preexistente vía GSM o GPRS. Es decir, es necesario plantear una solución que tenga en cuenta la portabilidad de los sensores, su adecuación al ambiente forestal de la región y una solución de conectividad simple.

Asimismo, en el marco de los sistemas embebidos se encuentran los sistemas críticos cuyo mal funcionamiento o fallo pueden resultar en pérdida de vidas humanas, equipos, instalaciones o daños medio ambientales severos [6]. Por ello, uno de los ámbitos de mayor utilización de sistemas críticos es el ferroviario.

Una normativa importante para el desarrollo de software crítico ferroviario es la norma EN 50128 [7]. En la misma se trata la calidad del software en términos del ciclo de vida del desarrollo, especificando los procedimientos y requisitos técnicos. Abarca los aspectos de seguridad en cinco niveles desde el 0 al 4, de acuerdo a la gravedad de las consecuencias del fallo, siendo el nivel 4 el que tiene asociado una mayor cantidad de técnicas y características a cumplir. A estos niveles de seguridad se los conoce con la sigla en inglés SSIL derivada de la expresión *software safety integrity level*.

En particular, en el apartado 7.4 de la norma EN 50128 se especifican las buenas prácticas al construir la arquitectura del software en términos de actividades, documentación, especificación integral de

cada módulo de la arquitectura y uso de buenas prácticas de programación.

En este sentido, los patrones de arquitectura que cumplen con las buenas prácticas antes mencionadas pueden ser complejos [8] y, a su vez, disminuir aspectos relacionados con la calidad del producto software, especialmente la mantenibilidad [9].

El propósito de esta línea de trabajo es, por tanto, desarrollar procedimientos aplicables a la construcción de sistemas críticos, así como su verificación y validación en ensayos junto con la Autoridad Ferroviaria Nacional, teniendo en cuenta el cumplimiento de la normativa EN 50128.

Ingeniería de Software para Entrega Continua

Tal y como lo describe la segunda ley de Lehman [10], el software necesita ser cada vez más complejo para satisfacer las necesidades de los usuarios. Esto, sumado a la importancia de detectar errores en la etapa de captura de requerimientos por sus costos [11], hace a la validación de los requerimientos una etapa crítica en el desarrollo de las aplicaciones [12].

Asimismo, las empresas regionales, como por ejemplo las empresas yerbateras o agrícolas tienen una estructura jerárquica tradicional, donde se identifican necesidades para gestión y seguimiento de tareas relacionadas a la Ingeniería de Software [13]. Estas empresas en su mayoría cuentan con una dirección que toma las decisiones, el cual conoce el negocio, pero no cuenta con el tiempo o la heurística suficientes para manifestar sus necesidades.

En este sentido, los desarrollos de metodologías ágiles representan un avance en la manera de construir sistemas; usando métodos, como, por ejemplo: el prototipado, las historias de usuario y los casos de uso. Esto hace posible la entrega temprana de valor, la respuesta rápida a los cambios y la colaboración constante del equipo de trabajo con los clientes [14][15].

Entonces, haciendo foco en la gestión de los requerimientos, es posible adaptar técnicas ágiles las cuales tendrán un fuerte impacto positivo [16][17], especialmente en PYMEs con problemas al momento de la gestión de sus requerimientos.

Las herramientas y prácticas genéricas deben ser seleccionadas y adaptadas para la empresa, tamaño y dominio específico. Y así, reducir los problemas causados por el no alineamiento tales como requisitos implementados incorrectamente, retrasos y esfuerzos desperdiciados [18].

Desde otro punto de vista, en el ámbito de fábricas de software se busca llevar adelante el desarrollo de las soluciones software con un enfoque evolutivo. Esto es cubierto por el enfoque de la ingeniería del software conocido como Entrega Continua de Software, en inglés Continuous Delivery (CD). Este es un enfoque en el cual los equipos mantienen la producción de software en ciclos cortos de tiempo, asegurando que el producto pueda ser lanzado de manera fiable en cualquier momento [19]. El objetivo es poder lanzar a producción un producto software libre de defectos “*con solo apretar un botón*” [20].

Uno de los principales desafíos de este enfoque es la calidad del producto software resultante. Ésta puede disminuir, dado que, al realizarse los despliegues del sistema con mayor frecuencia, aparecen más defectos en el producto. Por tanto, es esencial desarrollar un enfoque de priorización de los diferentes aspectos en la calidad del producto software, teniendo en cuenta la forma de trabajo actual de las empresas de desarrollo software.

Líneas de investigación y desarrollo

En la línea de Ingeniería de Software para Sistemas Embebidos y Sistemas Críticos:

- Diseñar y construir el prototipo funcional de sensores para la detección de focos de incendio y la construcción del software para la gestión de la red de sensores.

- Diseñar y construir un prototipo de antena Gateway LoRaWAN para la recepción de alarmas por parte de los sensores, que facilite la comunicación en entornos sin otro tipo de conectividad, así como la visualización de las alarmas.
- Realizar pruebas de concepto para la construcción de arquitecturas software y diseños específicos con patrones orientados a sistemas críticos de altas prestaciones.
- Planificar y ejecutar una Revisión Sistemática de la Literatura para identificar los estudios empíricos donde se verifican diferentes arquitecturas software de sistemas críticos y se comprueba el nivel de SSIL alcanzado.

En la línea de Ingeniería de Software y Entrega Continua:

- Utilizar el modelo de procedimientos para la gestión de requerimientos en PYMEs yerbateras a partir de técnicas ágiles.
- Desarrollar un modelo de madurez para el diseño y ejecución de pruebas continuas de software, especialmente orientado a flaky test y crossbrowsing testing.

Resultados obtenidos

El grupo de investigación es de reciente formación, por lo cual se enumeran los resultados de este último año. A continuación se indican:

En la línea de Ingeniería de Software para Sistemas Embebidos y Sistemas Críticos:

- En [21] se desarrolló un conjunto de procedimientos para la gestión de requisitos software en proyectos de sistemas críticos ferroviarios cumpliendo las buenas prácticas de la normativa EN 50128.
- En [22] se diseñó una prueba de concepto para arquitecturas software de sistemas críticos y se evaluaron los resultados en cuanto a complejidad y características del código fuente

resultante. De esta manera se pudo obtener un resultado comparable entre los diferentes patrones de arquitectura para software crítico, el nivel de seguridad asociado, la complejidad de su construcción y el grado de mantenibilidad resultante.

Asimismo, se está trabajando en el desarrollo de una red de sensores para la lectura automática de medidores eléctricos con comunicación LoRaWAN en el marco de un Acuerdo de Trabajo entre la Facultad de Ciencias Exactas y Naturales y Agrimensura (FACENA- UNNE) y la Dirección Provincial de Energía de la provincia de Corrientes [23].

En la línea de Ingeniería de Software para la Entrega Continua:

- En [24] se realizó una encuesta a más de 120 empresas y equipos de trabajo de todo el mundo para identificar el conjunto de problemas que afectan a la calidad del software en las disciplinas de entrega continua y pruebas continuas comparándolo con lo obtenido en Revisiones Sistemáticas de la Literatura anteriores
- En [25] se realizó una Revisión Sistemática de la Literatura para identificar las estrategias de personalización en las técnicas ágiles asociadas con la gestión de requisitos software.

Formación de recursos humanos

En el Grupo de Investigación en Innovación en Software y Sistemas Computacionales (GISSC) están involucrados 5 docentes investigadores, 1 becario de investigación de pregrado, 1 tesista de doctorado y 3 tesistas de maestría. Tres alumnos de la carrera están realizando sus proyectos finales vinculado a estos temas.

La tesis de maestría seleccionada para ser presentada llevó el título: “Integración de múltiples herramientas de software open source para la gestión y análisis de

rendimiento de clúster en ambiente HPC”, fue realizada por el Esp. Leopoldo José Ríos y dirigida por el Dr. Fernando Gustavo Tinetti.

El objetivo de la tesis fue el estudio, diseño e implementación de un sistema informático que integre múltiples herramientas de software Open Source para el sistema operativo Linux permitiendo la gestión global de recursos computacionales disponibles en un ambiente de Clúster HPC. El nivel de gestión que se desea obtener con esta aplicación, permitirá a los responsables operativos del Clúster, poder evaluar condiciones de usabilidad y planificación estratégica.

Los datos a obtener, provienen de los sistemas operativos que controlan el funcionamiento de la infraestructura general, determina la visualización del estado en tiempo real de Memoria RAM, Procesadores, estados de cachés, Consumo de la red de interconexión, entre otros. Serán utilizados también, programas como ‘perf’, que miden el rendimiento que tiene una aplicación mientras es ejecutada. Los resultados obtenidos con ‘perf’ serán almacenados e integrados a aquellos obtenidos desde el sistema operativo.

El almacenamiento de los datos, se hará sobre una base de datos de series de tiempo, con la herramienta RRDtool específicamente, a los efectos de poder elaborar y organizar información numérica a lo largo del tiempo. La misma herramienta nos servirá tanto para el proceso de registración, como para el proceso de generación de gráficos sobre consumos y usos que se consideren útiles para la gestión y análisis de rendimiento del clúster.

El tratamiento y gestión de los datos obtenidos, permitirá estudiar la demanda de equipamiento en un determinado período de tiempo, por ejemplo, un (1) año hacia atrás. Será posible la selección de criterios de búsqueda, ejemplo: uso de memoria RAM, uso de CPU, y los programas

utilizados en ese hecho, o usuarios del sistema involucrados.

Desarrollo y resultado.

La actividad de desarrollo fue solapada con actividades de gestión propias del clúster, del instituto tomado como caso de estudio de este trabajo. En el ámbito del instituto, la premisa siempre fue soportar el sistema de gestión de trabajos del clúster, para que los trabajos fluyan en el tiempo. La autenticación de usuarios fue desarrollada con técnicas RBAC (control de acceso basado en roles), con acceso a la ejecución de tareas por encima del acceso a objetos, es altamente aceptable y ha sido demostrada su efectividad. La plataforma generada con el framework “CakePHP”, con un despliegue modelo vista controlador, constituye la base de funcionamiento del sistema, y es ampliable a tantas funcionalidades como requerimientos de los usuarios exista.

Las vistas definidas en la aplicación, es el resultado de procesos internos que se disparan en función a la lógica de negocio impuesto por los permisos, asociados a roles, y éstos a usuarios. La aplicación del concepto RBAC se ha dado de manera muy simple, la experiencia indica que es necesario concentrarse en la identificación de roles, en la declaración y definición de permisos, y en la asignación a las cuentas de usuario de los roles establecidos, sin importar la cantidad de usuarios que sean. La herramienta Torque PBS se integra a la aplicación propuesta, como el módulo de gestión de trabajos en modalidad batch del clúster. Se encarga de la organización, clasificación, ejecución y finalización de trabajos de usuario. Provee una serie de comandos que hacen de interface al ser ejecutados, informan en tiempo real el estado de los trabajos de usuario; lo cual registrado por la aplicación para reunir información, registrarla y difundirla al usuario bajo demanda.

En lo referente a la administración de datos, se ha logrado gestionar datos de usuario como ser scripts, historiales de

trabajos ejecutados, con la herramienta de base de datos relacional MySQL. Los datos de monitoreo en línea se gestionan con la herramienta RRDtool, en la visualización de los distintos estados del sistema. Es posible demostrar que las herramientas se complementan perfectamente bien, sin observar interferencias.

Publicaciones relacionadas a la tesis.

Año 2017: V Jornadas de Cloud Computing & Big Data, desarrolladas en la Facultad de Informática de la UNLP, del 26 al 30 de junio de 2017. Trabajo “Acceso, Monitorización y Asistencia para el Desarrollo en Clusters para HPC: posible utilización en múltiples clusters y clouds”.

Año 2016: "Round Robin Data Bases for Performance Evaluation of High Performance Applications and Clusters", Fernando G. Tinetti, Leopoldo J. Rios, The 2016 International Conference on Grid, Cloud, and Cluster Computing (GCC'16), July 25-28, 2016, Monte Carlo Resort, Las Vegas, USA., Eds. Hamid R. Arabnia, Ashu M. G. Solo, Fernando G. Tinetti, ISBN: 1-60132-436-7, CSREA Press, pp 45-48.

Año 2015: “Instalación, Configuración y Evaluación de un Cluster HPC en una Nube Pública”, Fernando G. Tinetti, Leopoldo J. Rios, III Jornadas de Cloud Computing & Big Data, 29 de junio al 3 de julio de 2015, Fac. de Informática de la UNLP.

Conclusiones.

El enfoque de este trabajo de tesis fue elaborar una herramienta de gestión de clúster HPC basada en código abierto, para lo cual se ha incorporado otras herramientas de software que propician esta idea. El diseño previsto permite la interconexión entre las herramientas de manera muy simple, con la posibilidad de reemplazo de alguna de ellas y escasa re-configuración posterior. Es posible su migración a cualquier plataforma operativa: RedHat, Ubuntu, CentOS, y de gestión de datos como PostgreSQL. El resultado se pone a disposición de la comunidad para

permitir fácilmente la programación de tareas a ser desplegada por usuarios localizados en Internet. La programación de tareas requiere el conocimiento de programación PHP y MySQL para gestión de datos.

La aplicación permite gestionar equipos de computación configurados en clúster HPC, como un todo, o como partes de un todo. En el escenario del caso de estudio, el ámbito es el clúster completo: todos los nodos configuran colas de trabajo, que se gestionan desde la misma aplicación. Quizá en otros escenarios, determinen el uso de una parte del clúster, entonces la aplicación puede ajustarse específicamente a esos recursos, que luego los usuarios darán uso.

Referencias

- [1] Heath, Steve. *Embedded systems design*. Elsevier, 2002.
- [2] Jamieson, Peter. "Arduino for teaching embedded systems. are computer scientists and engineering educators missing the boat?." *Proc. FECS 2010* (2010): 289-294.
- [3] Minaverri, Clara María. La regulación de los incendios forestales en Argentina y Uruguay. *Ars Boni et Aequi*, 2016, vol. 12, no 1.
- [4] Loureiro, Maria L.; Allo, Maria. Los Incendios Forestales Y Su Impacto Económico: Propuesta Para Una Agenda Investigadora. *Revista Galega de Economía*, 2018, vol. 27, no 3, p. 129-142.
- [5] Oviedo, Byron, et al. Redes inalámbricas de sensores para detección temprana de incendios forestales. *Revista Científica Ciencia y tecnología*, 2019, vol. 18, no 20.
- [6] EN 50126. Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS). 2005.
- [7] EN 50128. Railway applications - Communication, signalling and processing systems - Software for railway control and protection systems. 2011.
- [8] W. Wu y T. Kelly, «Safety tactics for software architecture design», en *Proceedings of the 28th Annual International Computer Software and Applications Conference*, 2004. COMPSAC 2004., 2004, pp. 368–375.
- [9] L. Bass, P. Clements, y R. Kazman, *Software architecture in practice*. Addison-Wesley Professional, 2003.
- [10] M. Lehman, "On Understanding Laws, Evolution and Conservation in the Large Program Life Cycle", *J. of Sys. and Software*, vol. 13, pp. 213-221, 1980.
- [11] S. Maalem and N. Zarour, "Challenge of validation in requirements engineering", *Journal of Innovation in Digital Ecosystems*, vol. 3, no.1, pp. 15-21, 2016.
- [12] G. Kotonya and I. Sommerville, *Requirements engineering*. Chichester: John Wiley & Sons, 1998.
- [13] Plan Estratégico para el Sector Yerbatero – Yerba Mate Argentina", *Yerbamateargentina.org.ar*, 2016.
- [14] A. Sillitti and G. Succi, *Requirements Engineering for Agile Methods*. In: Aurum A., Wohlin C. (eds) *Engineering and Managing Software Requirements*. Berlin, Heidelberg: Springer, 2005.
- [15] I. Inayat, S. Salim, S. Marczak, M. Daneva and S. Shamshirband, "A systematic literature review on agile requirements engineering practices and challenges", *Computers in Human Behavior*, 2014.
- [16] S. Dragicevic, S. Celar and L. Novak, "Use of Method for Elicitation, Documentation, and Validation of Software User Requirements (MEDoV) in Agile Software Development Projects", *CICN* 2014.
- [17] One, V. "State of agile development survey results", 2017.
- [18] E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek and R. Feldt, "Challenges and practices in aligning requirements with verification and validation: a case study of six companies", *Empirical Software Engineering*, vol. 19, no. 6, pp. 1809-1855, 2013.
- [19] L. Chen, "Continuous Delivery: Huge Benefits, but Challenges Too" in *IEEE Software* 03/2015. V. 32(2).
- [20] J. Humble and D. Farley. "Continuous delivery: reliable software releases through build, test, and deployment

automation”, 1st ed. Boston, US: Pearson Education, 2010.

- [21] Pinto Luft, Cristian: Construcción de una metodología de gestión de requerimientos software y desarrollo de un ecosistema de herramientas de acuerdo con la norma EN 50128. Aplicación en el desarrollo de un prototipo para la Autoridad Ferroviaria Nacional. Maestría en Tecnologías de la Información FaCENA – UNNE. Tesis de Maestría
- [22] Acevedo Duprat Joaquín, Irrazábal Emanuel, Evaluación de la calidad en patrones de arquitecturas software para sistemas críticos ferroviarios: un enfoque basado en la mantenibilidad, XXV Congreso Argentino de Ciencias de la Computación. Río Cuarto (Argentina). Octubre de 2019.
- [23] Acuerdo de Cooperación entre la FACENA y la Dirección Provincial de Energía de Corrientes para el desarrollo de un prototipo de comunicación entre medidores domiciliarios con colector en subestación y centro para el registro remoto del consumo y otros parámetros en tiempo real. Res. 0766/19 CD Facena.
- [24] Mascheroni, Maximiliano, Irrazábal, Emanuel Carruthers, Juan, Pinto, Juan. Rapid Releases and Testing Problems at the industry: A survey. XXV Congreso Argentino de Ciencias de la Computación. CACIC 2019. Río Cuarto, octubre 2019.
- [25] Pereyra Coimbra, Rodrigo, Carruthers, Juan, Pinto, Juan, Irrazábal, Emanuel. Personalización de técnicas ágiles en el desarrollo de software para la obtención de requisitos de calidad en Pymes: una revisión sistemática de la literatura. XX Simposio Argentino de Ingeniería de Software (ASSE 2019)-JAIIO 48. Salta, Septiembre de 2.019